Robust Video Coding Techniques

for

Wireless Video Communications

Final Report on SBIR Phase I Project (DAAB07-96-C-G009)

Vector Vision, Inc.

117 ATLSS Drive

Imbt Laboratory

Bethlehem, PA 18015-4729

Abstract — **The objective of this SBIR project is to develop a video coding technique that is robust for wireless video communications in terms of video quality degradation as channel noise increases. Phase I of the project is a feasibility study of several innovative error-handling techniques based on the framework of vector transform coding, which has been shown to provide much better compression performance than the standard techniques for still image and video compression [1]. As a result of the phase I project, three different index assignment techniques for vector quantization in vector wavelet domain have been studied. The three techniques are (1) optimal index assignment without knowing the code-vector distribution; (2) optimal index assignment with known code-vector distribution; and (3) joint optimization of index assignment and codebook design. Compared with the standard video coding method with and without error correction coding, all three new techniques have shown much more robust performance in a wide range of channel noise levels. Among the three new techniques, we have concluded that the second one is the best, which provides better performance than the first one and about the same performance as the third one but with lower complexity.**

I. Introduction

Video communication has become an increasingly important tool in many military and civilian operations. Because the raw data rate of a digital video signal is usually much higher than the capacity of any transmission channel, video compression has to be used in video communications. Several international standards have been established for video compression. Although the standard video compression techniques can be used in many applications, they are not suitable for wireless video communications. The reason is that a compressed bit-stream using one of the standards is very sensitive to channel noise and a wireless channel has much higher noise level than a wired channel. The conventional wisdom is to use error-correcting codes for the compressed bit-stream. There are several problems with such a conventional approach. First of all, the transmission data rate increases because the error-correcting codes introduce redundancy. Secondly, the complexity of the video communication system increases due to using an error-correcting encoder and decoder, which means a higher cost. Thirdly, an important characteristic of wireless communication channels is that the noise level varies in a very wide range. If an error-correcting code was used to satisfy the worst case noise condition, then a lot of redundancy would have to be added into the compressed bit-stream. This means higher transmission data rate or lower information data rate. If only the average noise condition was considered in using the error-correcting code, the bit-stream would be completely damaged if the noise becomes worse than the average condition. Due to all these problems, application of error correcting codes for wireless video communications is not a very good approach. Re-transmission is a method commonly used in computer communications for error handling. However, because the latency for video communications cannot be too long, re-transmission is not feasible either. Ideally, we would like to use a video coding technique that adapts to the channel

condition. When the channel noise is small, a high quality video signal can be reconstructed from the compressed bit-stream. When the channel noise increases, degradation of video quality is proportional to the noise level, instead of a sudden quality drop. Figure 1 illustrates the problems with the conventional approach and what an ideal situation should be. The horizontal axis is the channel noise level. The vertical axis is the received video quality due to both quantization and channel noise. Assuming a fixed transmission data rate, a fixed channel bandwidth, and a fixed transmission signal power, curves 1, 2, and 3 illustrate how received video quality changes with the channel noise level when the conventional approach is used to design the wireless video communication system according to the best, the average, and the worst noise conditions respectively. If the system is designed according to the best noise level, quantization distortion can be made small and the information data rate is the same or very close to the transmission data rate. Because no error correcting capability is added, video quality drops quickly when the channel noise level increases a little as shown by curve 1. If the system is designed according to the worst noise condition, then the video quality is poor even when the channel noise is small because quantization distortion has to be large in order to lower the information data rate for adding a strong error correcting code. As shown by curve 3, received video quality keeps at a constant for a wide range of channel noise level. But the constant video quality is very low to start with, due to a large quantization distortion. On the other hand, curve 4 in Figure 1 shows the performance of an ideal system. When the channel noise is small, the video quality is very high. When channel noise increases, the video quality degrades gradually. Therefore, a technical challenge of wireless video communications is to develop a video coding technique for robust transmission of the compressed bit-stream so that the abrupt video quality degradation can be avoided. Such a video coding technique will allow the user to enjoy all the benefits of having digital video communications without the problem of abrupt quality degradation due to channel noise.

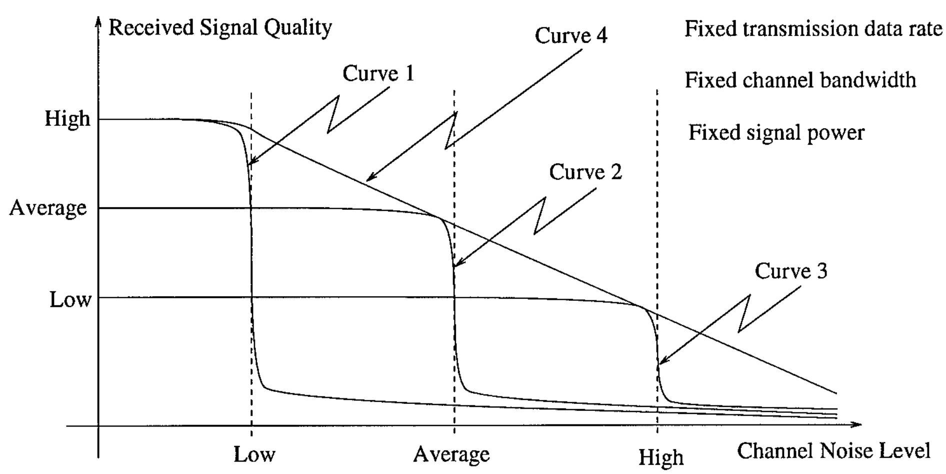

Figure 1: Received signal quality versus channel noise level

This SBIR project is to develop such a robust video coding technique. In the next section of this report, three new index assignment techniques studied in this project are outlined. In section III, accomplishments of this project are reported. In section IV, detailed technical results from the studies of this project are presented. Finally, section V concludes

this report and points out some future R&D directions to bring the new technology to practical applications.

II. Three Index Assignment Techniques for Vector Wavelet Coding

Any video compression technique can be modeled as a three-stage process (with or without feedback) as shown in Figure 2. The first stage is signal processing that converts a video signal from its original form to a different domain in which the signal is better prepared for the second stage operation. The second stage is quantization that is a many-to-one mapping. The third stage is lossless coding of the quantization results to achieve further compaction.

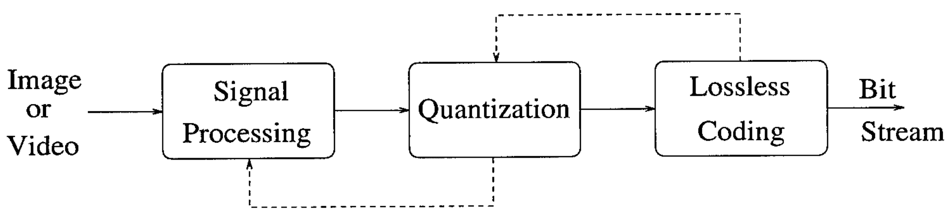

Figure 2: Model of a video compression technique

In this project, vector wavelet coding (VWC) is used as the basic framework because it provides good compression performance. This scheme can be described as the following procedure:

1. a video frame is sub-sampled into $M \times M$ sub-frames;
2. a wavelet transform is applied to each sub-frame;
3. the corresponding wavelet coefficients of the $M \times M$ sub-frames are grouped into a vector of dimension $M \times M$;
4. VQ is applied to each vector.

This project is to explore the features of the VWC technique for robust video communications. VQ in the vector wavelet domain is basically a mapping of a k-dimensional vector to the closest code-vector in a codebook according to a distortion measure. The index of the closest code-vector is transmitted to the receiving end. Design of the codebook is to find $N = 2^B$ code-vectors so that the average distortion is minimized. Usually, a B-bit index is assigned to each code-vector in an arbitrary way. This project is to study how to change the arbitrary index assignment to an optimal index assignment so that the error in the received index causes as little distortion as possible in the reconstructed video. The following three techniques have been studied in this project.

1. *Optimal index assignment for a given codebook without knowing code-vector distribution*

   Let $\mathbf{C} = \{\mathbf{c}_n, n = 0, 1, 2, \cdots, N-1\}$ be a codebook of N code-vectors. In VQ encoding, an input vector $\mathbf{x}$ is compared with the N code-vectors according to a distortion measure such as the mean squared error (MSE) as follows:

$$MSE(n) = \frac{1}{k}\sum_{i=0}^{k-1}(x_i - c_{n,i})^2$$

for $n = 0, 1, 2, \cdots, N-1$. Then the code-vector with the minimum MSE (closest to the input vector) is chosen to represent the input vector. Let $\mathbf{c}_m$ be such a code-vector, i.e., $MSE(m) \leq MSE(n)$ for $n = 0, 1, 2, \cdots, N-1$. The index of the closest code-vector, m, is transmitted through a communication channel. If $N = 2^B$, then B bits are needed to code the index m. At the receiving end, decoding is simply a table look-up by using the received index m to read out the code-vector $\mathbf{c}_m$ from the codebook, which is the closest representation of $\mathbf{x}$. The problem now is that the transmitted B-bit index m may be changed by the channel noise and the received index becomes m' which may be one bit different from m. Since assignment of indexes $\{0, 1, 2, \cdots, N-1\}$ to the code-vectors in $\mathbf{C}$ is usually arbitrary. The code-vector $\mathbf{c}_{m'}$ read out of the codebook at the receiving end can be very different from the code-vector $\mathbf{c}_m$ found to be the closest representation of the input vector $\mathbf{x}$ at the transmission end although the received index m' differs from the transmitted index m by only one bit. One of the technical objectives in this Phase I project is to find an index assignment method for the code-vectors in the codebook $\mathbf{C}$ such that the difference between any two code-vectors is proportional to the difference between the two indexes assigned to the two code-vectors. The difference between two code-vectors $\mathbf{c}_n$ and $\mathbf{c}_m$ is measured by the Euclidean distance between the two code-vectors defined as follows:

$$d_e(\mathbf{c}_n, \mathbf{c}_m) = \sqrt{\sum_{i=0}^{k-1}(c_{n,i} - c_{m,i})^2}$$

The difference between two indexes n and m is measured by the Hamming distance between the two indexes defined as follows:

$$d_h(n, m) = \text{the number of bits that are differ in } n \text{ and } m$$

Using these two error measurements, the optimal index assignment should ensure that, if $d_h(n, m)$ is small, then $d_e(\mathbf{c}_n, \mathbf{c}_m)$ is small. Without knowing the distribution of code-vectors, uniform distribution is assumed. The idea is to start with an arbitrary index assignment and adjust it toward our goal. More specifically, we have developed a computer software to implement the following procedure:

- Assign index "0" to a code-vector randomly chosen in the codebook since uniform distribution is assumed, and make this code-vector the first center code-vector.
- Find code-vectors that have close Euclidean distances with the center code-vector and assign to those code-vectors the indexes with Hamming distance of 1 relatively to the index of the center code-vector.

- Take a code-vector that has been assigned an index but has not been a center code-vector to be the next center code-vector and repeat the previous step.
- Stop when the index of the center code-vector has the highest Hamming weight of B.

2. *Optimal index assignment for a given codebook with a given probability distribution of the code-vectors*

   In real applications, usage of code-vectors in a codebook is usually non-uniform. Some of the code-vectors are used more often than others. The transmission error on the indexes of those frequently used code-vectors causes more distortions in the reconstructed video. Therefore, we should put more weight on those frequently used code-vectors so that the transmission errors of their indexes cause as little distortion as possible. The second technical objective of this project is to find an index assignment method that takes into consideration of the probability distribution of the code-vectors in addition to the Euclidean distance between them and the Hamming distance between their indexes.

   The computer software developed in the previous task is the starting point of this task. The only thing we need to add into the previous program is the probability distribution of code-vectors, which is usually non-uniform. This means that some code-vectors may be more important than others because they are used more frequently than others. With a modification to the previous procedure, we have developed another computer software to carry out the following process:

   - Assign index "0" to the code-vector with the highest probability in the codebook and make this code-vector the first center code-vector.
   - Find code-vectors that have close Euclidean distances with the center code-vector and assign to those code-vectors the indexes with Hamming distance of 1 relatively to the index of the center code-vector.
   - Take the code-vector with the highest probability among those that have been assigned indexes but have not been used as center code-vectors to be the next center code-vector and repeat the previous step.
   - Stop when the index of the center code-vector has the highest Hamming weight of B.

3. *Joint optimization of codebook and index assignment for a given source probability distribution*

   The previous two programs can find the best index assignment under the assumption that the codebook is given. There, the locations of the code-vectors are determined in codebook design to minimize the average distortion according to the source distribution without considering the channel noise. The best index assignment ensures that the minimum additional distortion is introduced if the indexes are changed by channel noise. The third technical objective of this project is to find a method for codebook design and index assignment that minimizes the total average distortion caused by both quantization and channel noise.

The idea is to jointly optimize the locations of the code-vectors and the assignment of indexes to the code-vectors at the time of designing the codebook. The conventional codebook design software is modified to take into consideration of channel noise. The modified codebook design software performs the following procedure:

- Find the centroid of the training set;
- Split the centroid into two code-vectors very close to the centroid;
- Use the LBG algorithm to find the optimal codebook with two code-vectors;
- Follow the splitting and optimizing procedure until the codebook of 2^{B-1} code-vectors is obtained;
- Split each code-vector in the codebook of 2^{B-1} code-vectors into two closely located code-vectors;
- Partition the training set into 2^B groups with each group associated with one and only one code-vector according to the nearest neighbor rule;
- Assign index zero to the group with the most number of training vectors and other indexes to other groups according to the rule of the Hamming distance being proportional to the Euclidean distance from the group zero;
- Refine the index assignment using the second largest group as the center without affecting the situation for the largest group;
- Refine the index assignment using other groups with an order of decreasing size until all groups are used;
- Move the location of each code-vector so that the total distortion caused by both quantization and index error is minimized;
- Re-partition the training set according to the new locations of the code-vectors;
- Repeat the index assignment and code-vector re-allocation procedure until re-partition does not change any group of training vectors.

III. Accomplishments of the Project

In this project, we have achieved the objective of studying the feasibility of vector wavelet coding for robust video communications with the following accomplishments and deliverables:

- Software to find the optimal index assignment for a given codebook without knowing code-vector distribution
  - Source code file name: method-1.c
  - Usage: method-1 input-codebook output-codebook bits top
- Software to find the optimal index assignment for a given codebook with a given probability distribution of the code-vectors

- Source code file name: method-2.c
- Usage: method-2 finame focname fcname Nr Nc Nf

- Software to perform joint optimization of codebook and index assignment for a given source probability distribution
  - Source code file name: method-3.c
  - Usage: method-3 finame foname

- Software to implement the Reed-Muller code for error correcting
  - Source code file names: rm-encode.c, rm-decode.c
  - Usage:
    rm-encode finame foname fsize
    rm-decode finame foname fsize

- Software to implement vector wavelet transform
  - Source code file names: fvwt.c, ivwt.c
  - Usage:
    fvwt finame foname fcname Nr Nc Nf
    ivwt finame foname fcname Nr Nc Nf level

- Software for video format conversion
  - Source code file name: ccir2sif.c
  - Usage: ccir2sif finame foname Nf Nr Nc format

- Software for calculating PSNR of a reconstructed video sequence
  - Source code file name: psnr.c
  - Usage: psnr foname fqname Nr Nc Nf pflag

- Results of applying MPEG-1 on the test sequence without an error correcting code
  - Data file names: mpeg1.10e-4, mpeg1.10e-3, mpeg1.10e-2
  - Figure numbers: Figures 6,7,8

- Results of applying MPEG-1 on the test sequence with a Reed-Muller code
  - Data file names: mpeg1ecc1.10e-4, mpeg1ecc1.10e-3, mpeg1ecc1.10e-2, mpeg1ecc2.10e-4, mpeg1ecc2.10e-3, mpeg1ecc2.10e-2
  - Figure numbers: Figures 9,10,11,12,13,14

- Results of applying the new techniques in the vector wavelet domain

– Data file names: method1.10e-4, method1.10e-3, method1.10e-2, method2.10e-4, method2.10e-3, method2.10e-2, vwc.no-error

– Figure numbers: Figures 15,16,17

IV. Technical Results

In this feasibility study, we have compared the performance of the new methods with that of the conventional approach. The procedure for evaluation and comparison is as follows:

- An MPEG-1 encoding software is used for compressing the test sequence at a bitrate of 1.58Mb/s. The compressed bit-stream is sent to a channel noise simulation program without error correcting coding. Three different bit error rates (BERs) (10e-4, 10e-3, and 10e-2) of channel noise have been applied. The output bit-stream of the channel noise simulation program is sent to the program for MPEG-1 decoding to reconstruct the test video sequence. The peak signal-to-noise-ratio (PSNR) of the reconstructed test sequence as compared with the original sequence is calculated. Figure 3 shows the process.

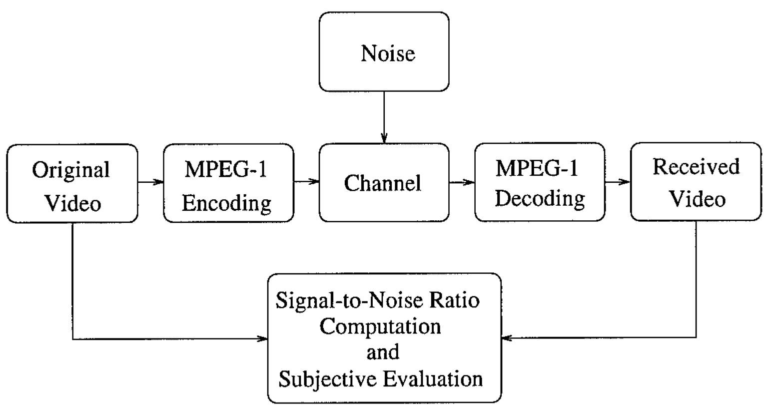

Figure 3: MPEG-1 without error correcting coding

- The same procedure as the previous one is performed, but the compressed bit-stream is sent to the channel noise simulation program after passing through a Reed-Muller encoding program. When the bit-rate of MPEG-1 encoding is maintained at 1.58Mb/s, the transmission bit-rate becomes 2.3Mb/s after Reed-Muller encoding. We have also obtained test data of keeping the transmission bit-rate about the same (1.64Mb/s) while MPEG-1 compression has to be at a lower bitrate. The same three BERs are used for channel noise. The output bit-stream of the channel noise simulation program is sent to the programs for error correcting decoding and MPEG-1 decoding to reconstruct the test video sequence. The PSNR of the reconstructed test sequence as compared with the original sequence is calculated. Figure 4 shows the process.

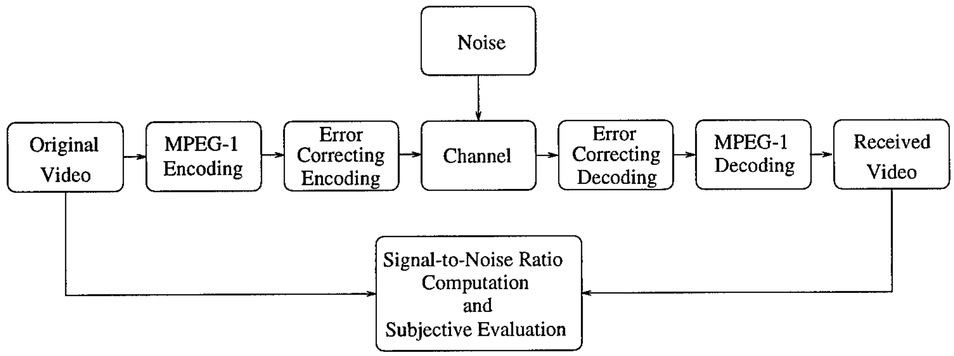

Figure 4: MPEG-1 with error correcting coding

- The three new methods are tested. All three methods share the same vector wavelet transform and its inverse. The difference between the first two new methods is the index assignment. The third new method differs from the first two by having a different set of codebooks designed by the joint optimization of the locations and the index assignment of the code-vectors. For all three new methods, the compressed bit-stream is sent to the channel noise simulation program without passing through the error correcting coding program. For all three new methods, the bit-rate is 1.58Mb/s. The same three BERs are used for channel noise. The output bit-stream of the channel noise simulation program is sent to the programs for inverse VQ and the inverse vector transform to reconstruct the test video sequence. The PSNR of the reconstructed test sequence as compared with the original sequence is calculated. Figure 5 shows the process.

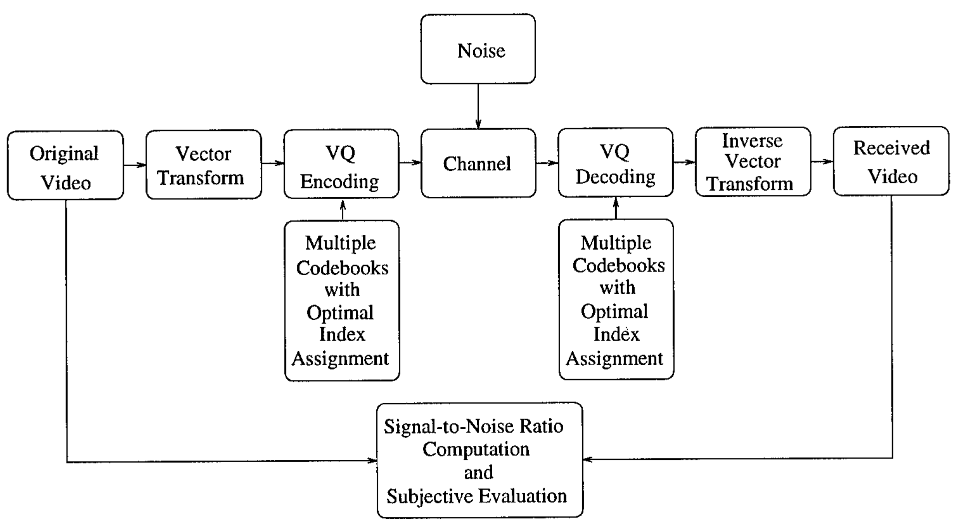

Figure 5: Process to test the three new methods

For MPEG-1 encoding and decoding, we have used a public domain MPEG-1 software from UC Berkely. We have used an MPEG-4 software to add channel noise to the compressed bit-stream.

A set of comparison plots have been generated. Figures 6,7,8 compare the second new

method with MPEG-1 in terms of PSNR at the same bit-rate of 1.58Mb/s and BERs of 10e-4, 10e-3, and 10e-2 respectively. In this comparison, MPEG-1 bit-streams are not protected by an error correcting code. It is clear from these three plots that, without protection of an error correcting code, the MPEG-1 bit-stream cannot be decoded even at BER of 10e-4 after frame 59 (PSNR drops to zero for the MPEG-1 curve). Because the channel noise simulation program does not introduce any channel errors for the first 1.5 seconds (45 frames), this means that the MPEG-1 bit stream is completely destroyed by the channel errors almost as soon as the channel errors are introduced and no video frames can be reconstructed as soon as the channel errors are present. On the other hand, the new method performs reasonably well with some PSNR degradation as BER increases from 10e-4 to 10e-3 and 10e-2 and the video sequence can still be reconstructed from the bit-stream.

Figures 9,10,11 compare the second new method with MPEG-1 in terms of PSNR at BERs of 10e-4, 10e-3, and 10e-2 respectively. In this comparison, MPEG-1 bit-streams are protected by an error correcting code and the transmission rate of the protected MPEG-1 bit-stream is 2.3Mb/s while the bit-rate for the new method is still 1.58Mb/s. These three plots show that, by adding an error correcting code to the MPEG-1 bit-stream (and transmission bit-rate increases to 2.3Mb/s), almost the entire MPEG-1 bit-stream can be decoded when BER is 10e-4 (PSNR drops to zero at the end of the sequence). When BER increases to 10e-3, MPEG-1 PSNR degradation starts before frame 60 and drops to zero before frame 125. As BER increases to 10e-2, the error correcting code does not help at all and the MPEG-1 curve becomes the same as that without error correcting code. Again, the MPEG-1 bit-stream is completely destroyed by the channel errors almost as soon as the channel errors are introduced and no video frames can be reconstructed as soon as the channel errors are present.

Figures 12,13,14 compare again the second new method with MPEG-1 in terms of PSNR at BERs of 10e-4, 10e-3, and 10e-2 respectively. In this comparison, MPEG-1 bit-streams are protected by the same error correcting code as the previous comparison but the transmission rate of the protected MPEG-1 bit-stream is 1.64Mb/s while the bit-rate for the new method is still 1.58Mb/s. The transmission bit-rate of MPEG-1 is lower than that in the previous comparison by compressing the video sequence more. The effect is that more quantization noise is introduced and relative channel error protection is increased. As shown in these three plots, when channel BER is 10e-4, the entire MPEG-1 bit-stream can be decoded. When channel BER is increased to 10e-3, the MPEG-1 curve becomes almost the same as the previous case with PSNR dropping to zero at a few frames later. However, when channel BER is increased to 10e-2, the error correcting code again does not help at all and the MPEG-1 curve becomes the same as that without error correcting code. Again, the MPEG-1 bit-stream is completely destroyed by the channel errors almost as soon as the channel errors are introduced and no video frames can be reconstructed as soon as the channel errors are present.

Figures 15,16,17 compare the relative performance of the new methods in terms of PSNR at a bit-rate of 1.58Mb/s and channel BERs of 10e-4, 10e-3, and 10e-2 respectively. Since the performance of the third new method is almost identical to that of the second new method, only two curves are compared with that without channel noise. As shown by these three plots, the relative performance gain of the second new method over the first new method is larger when channel BER is larger.

V. Conclusions and Future R&D

In this project, we have developed three new index assignment and codebook design methods for VQ in the vector wavelet domain. Extensive simulation has been performed to compare the new methods to the conventional approach of video compression plus error correcting coding. The conclusions from this study can be outlined as follows:

- The new methods are much more robust than the conventional methods in a wide range of channel noise conditions.
- The second new method provides a better performance than the first one and about the same as the third one. Because the second new method is simpler than the third one, we recommend it as the result of this feasibility study.
- The conventional video compression method without error correcting coding is too sensitive to channel errors as evidenced by the fact that, even at channel BER of 10e-4, the bit-stream is completely destroyed by the channel errors almost as soon as the channel errors are introduced and no video frames can be reconstructed as soon as the channel errors are present.
- At a channel BER of 10e-4, error correcting coding can improve the robustness of the conventional video compression method.
- At a channel BER of 10e-3, error correcting coding can delay the failure point of the conventional video compression method but reconstructed video quality is too low.
- At a channel BER of 10e-2, the conventional video compression method fails with or without error correcting coding.

The results of this Phase I project is very encouraging. We would like to implement the second new method in a prototype wireless video communication system. In the phase-II project, as an intermediate step of building a prototype system, we will demonstrate software based one-way video transmission over a wireless modem link. Two computers connected through a pair of wireless modems will be used to perform client/server functions. Compressed video sequences will be stored in the server computer. The decoding software will be installed on the client computer. The bit-stream of compressed video sequences will be transmitted through the wireless modem link from the server to the client which will decode and display the reconstructed video sequences. Because the decoding part of Vector Vision's technique is very simple, software only decoding is fast enough for real-time display of reconstructed video sequences. For one-way video, encoding does not have to be in real-time. Therefore, no special hardware is needed.

In the Phase II project, a hardware board will be designed to perform real-time capture and encoding of video signals at different levels of complexity. The decoding part can still be handled by software only. The prototype system will demonstrate not only good compression performance but also robust compression performance under various channel noise conditions. It will be an ideal system for wireless video communications, where high compression performance is required because the bandwidth and signal-to-noise ratio of wireless

channels are very limited and robust compression performance is also required because the channel noise varies within a wide range. The wireless video communication system using the results of this project will satisfy both requirements. It will provide the best video quality under any channel noise condition.

The end product of the phase-II project will be a set of dual-use technologies and a prototype system ready for integration into any digital wireless communication systems. For commercial applications, a possible product is a hardware board installed into laptop and desktop computers for wireless video conferencing. Another possible commercial product is a video cellular phone in a car. For military applications, a possible product is a pocket-sized transmitter/receiver with video/audio codec connected to a helmet with a small head-mounted camera and a pair of display glasses for video input and output and a microphone and an earphone for audio input and output.

References

[1] Weiping Li and Ya-Qin Zhang, "Vector-Based Signal Processing and Quantization for Image and Video Compression", Proceedings of the IEEE, Vol.83, No.2, February 1995, pp 317-335.

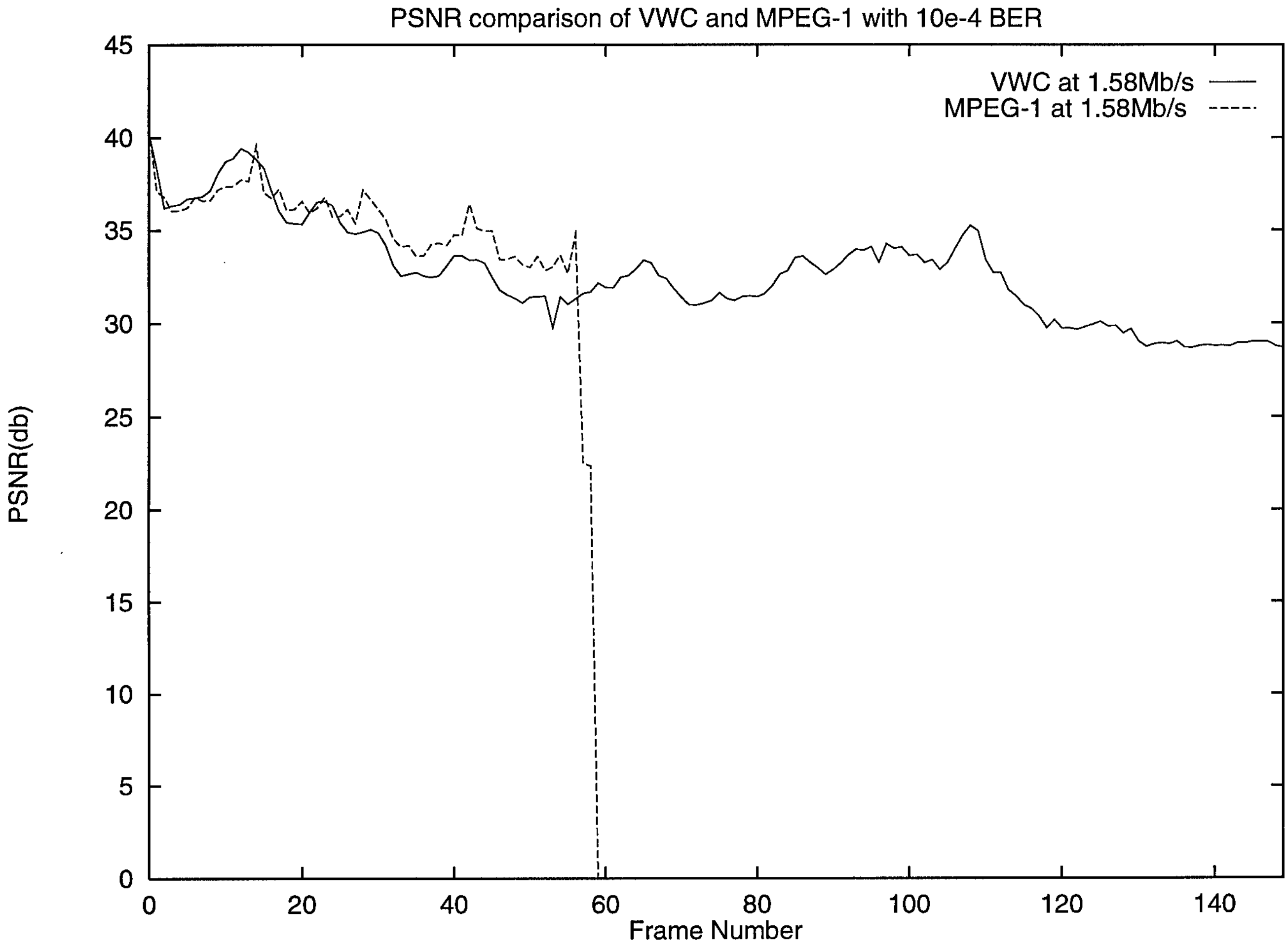

Figure 6: Comparison of VWC with MPEG-1 at 1.58Mb/s and 10e-4 BER

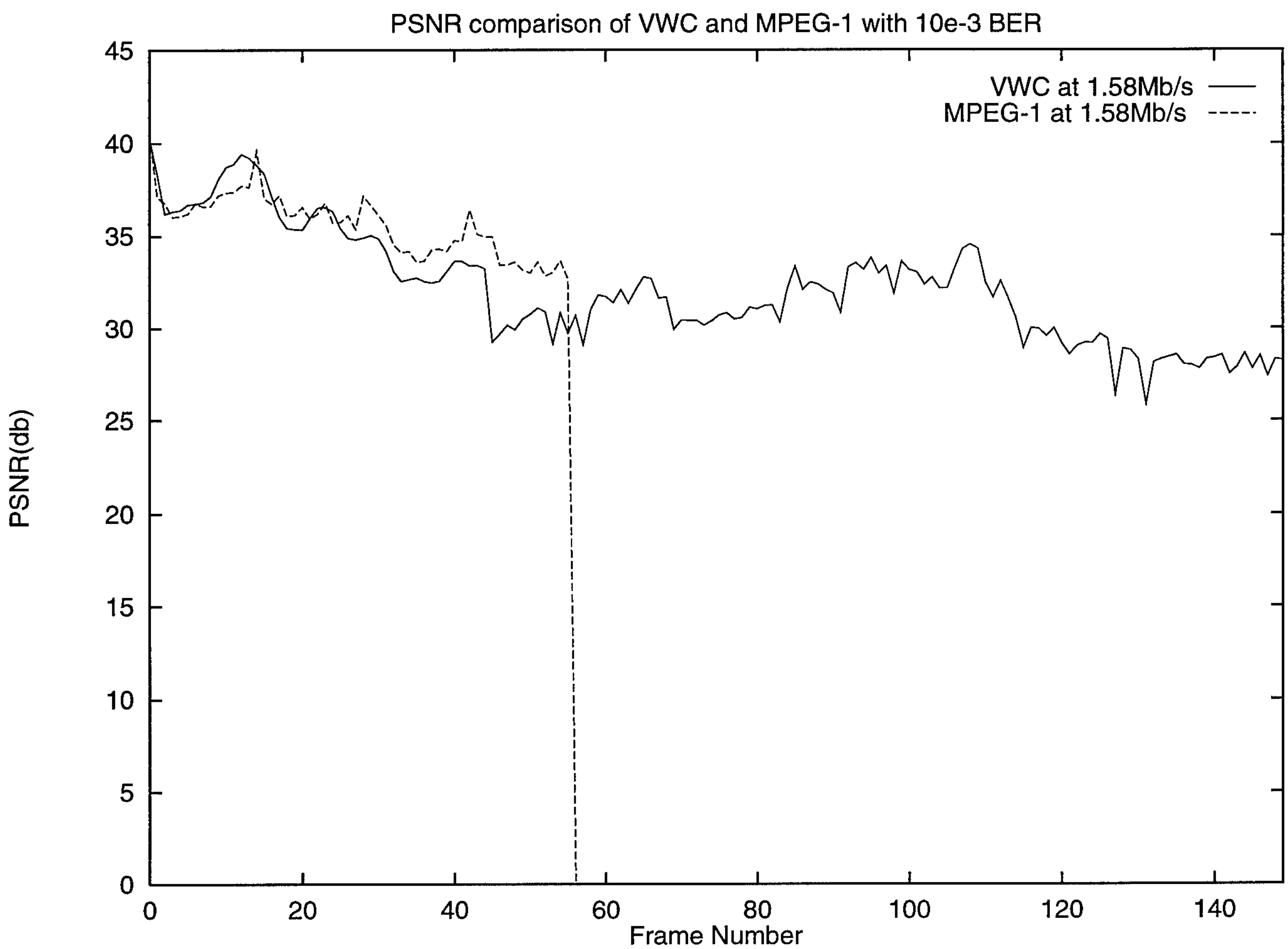

Figure 7: Comparison of VWC with MPEG-1 at 1.58Mb/s and 10e-3 BER

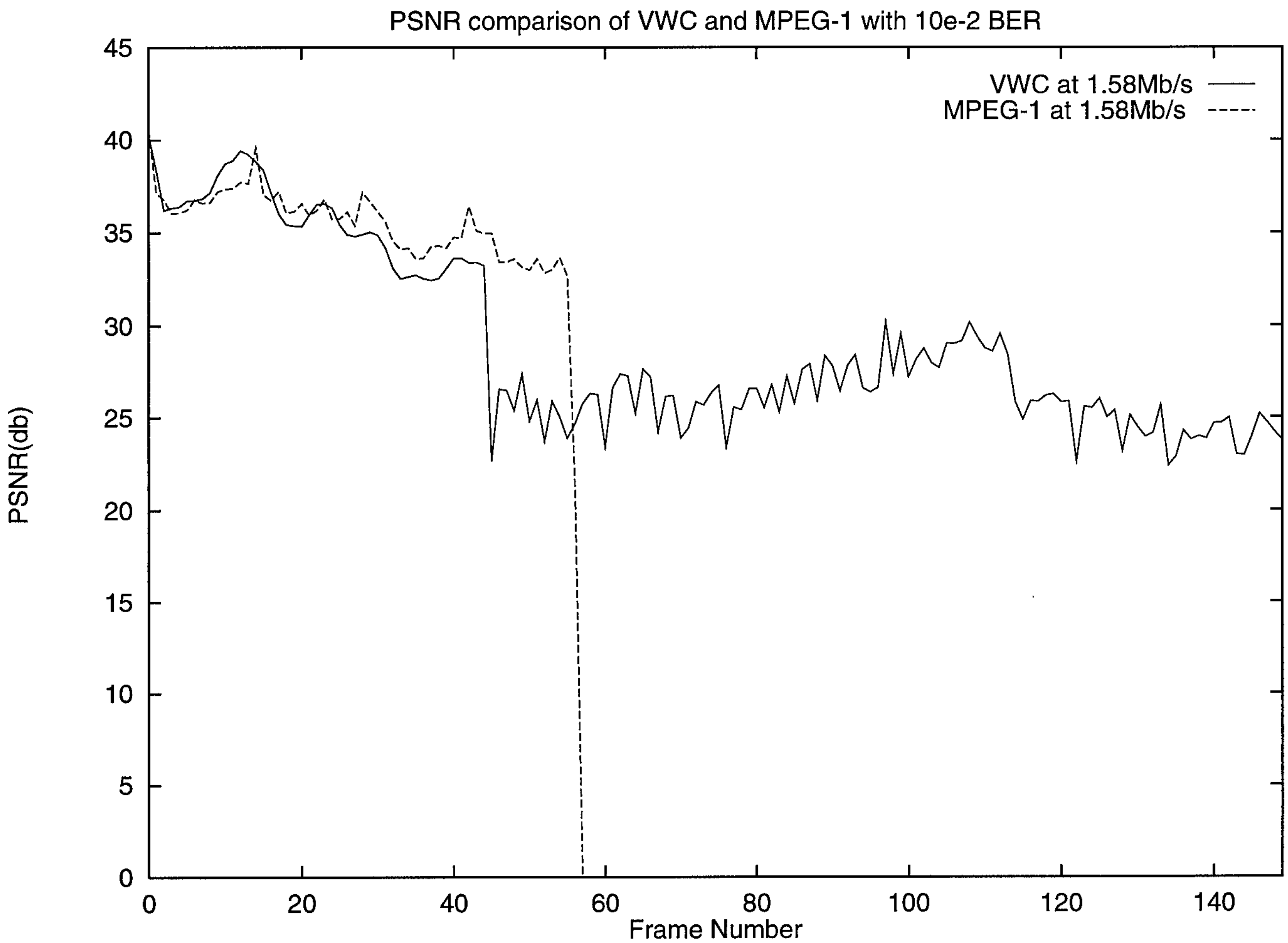

Figure 8: Comparison of VWC with MPEG-1 at 1.58Mb/s and 10e-2 BER

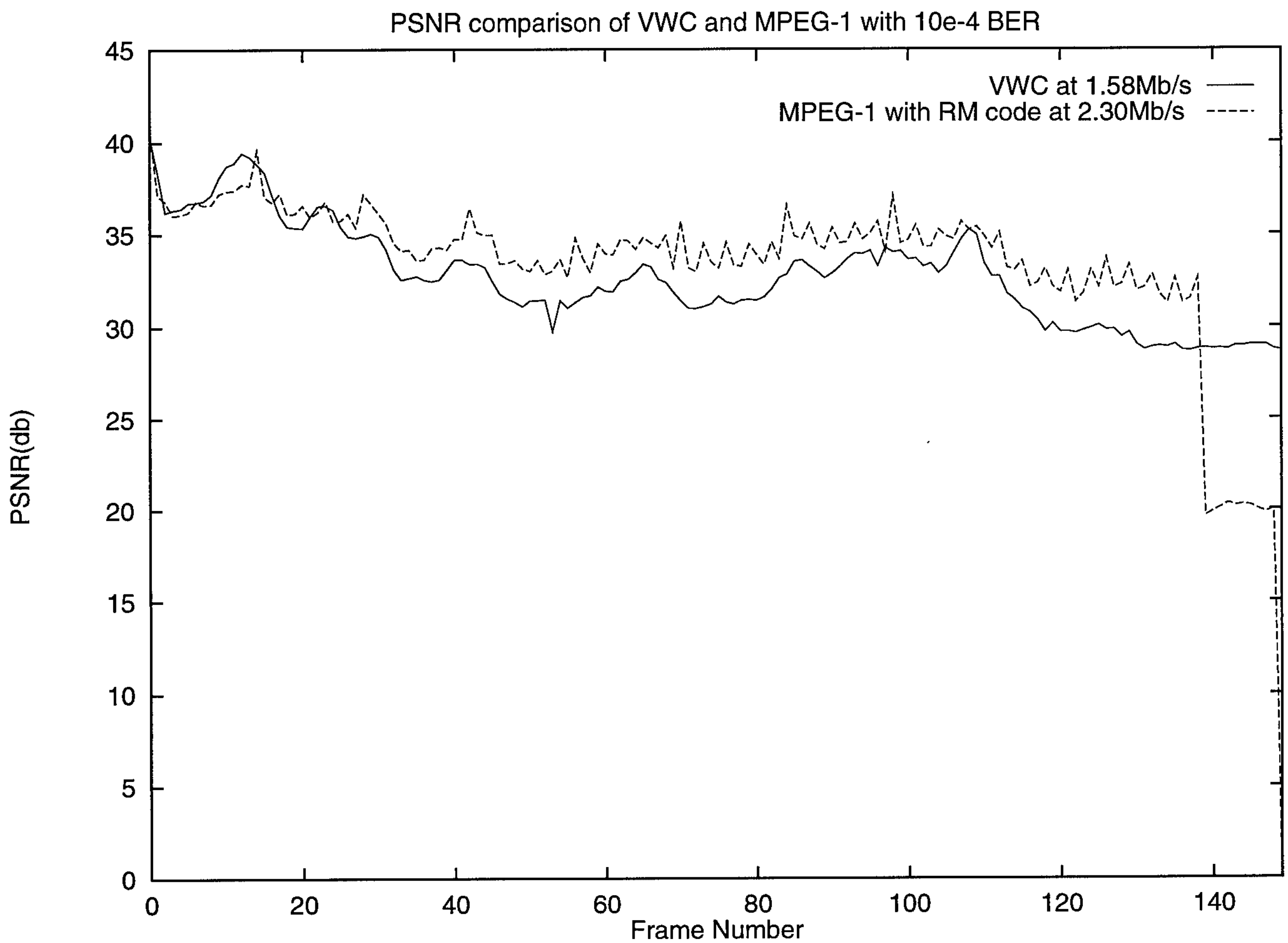

Figure 9: Comparison of VWC with MPEG-1 plus RM code at 2.3Mb/s and 10e-4 BER

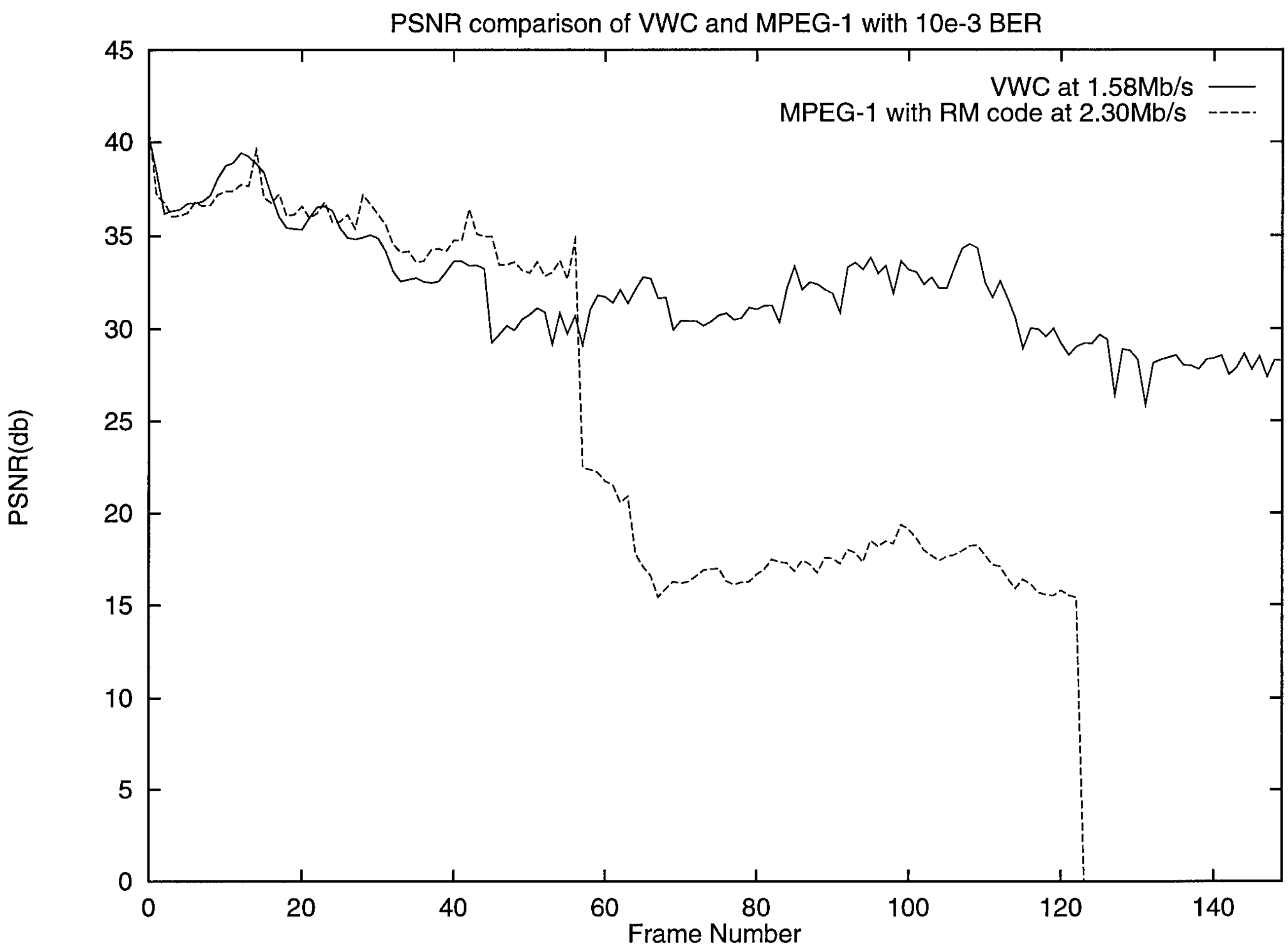

Figure 10: Comparison of VWC with MPEG-1 plus RM code at 2.3Mb/s and 10e-3 BER

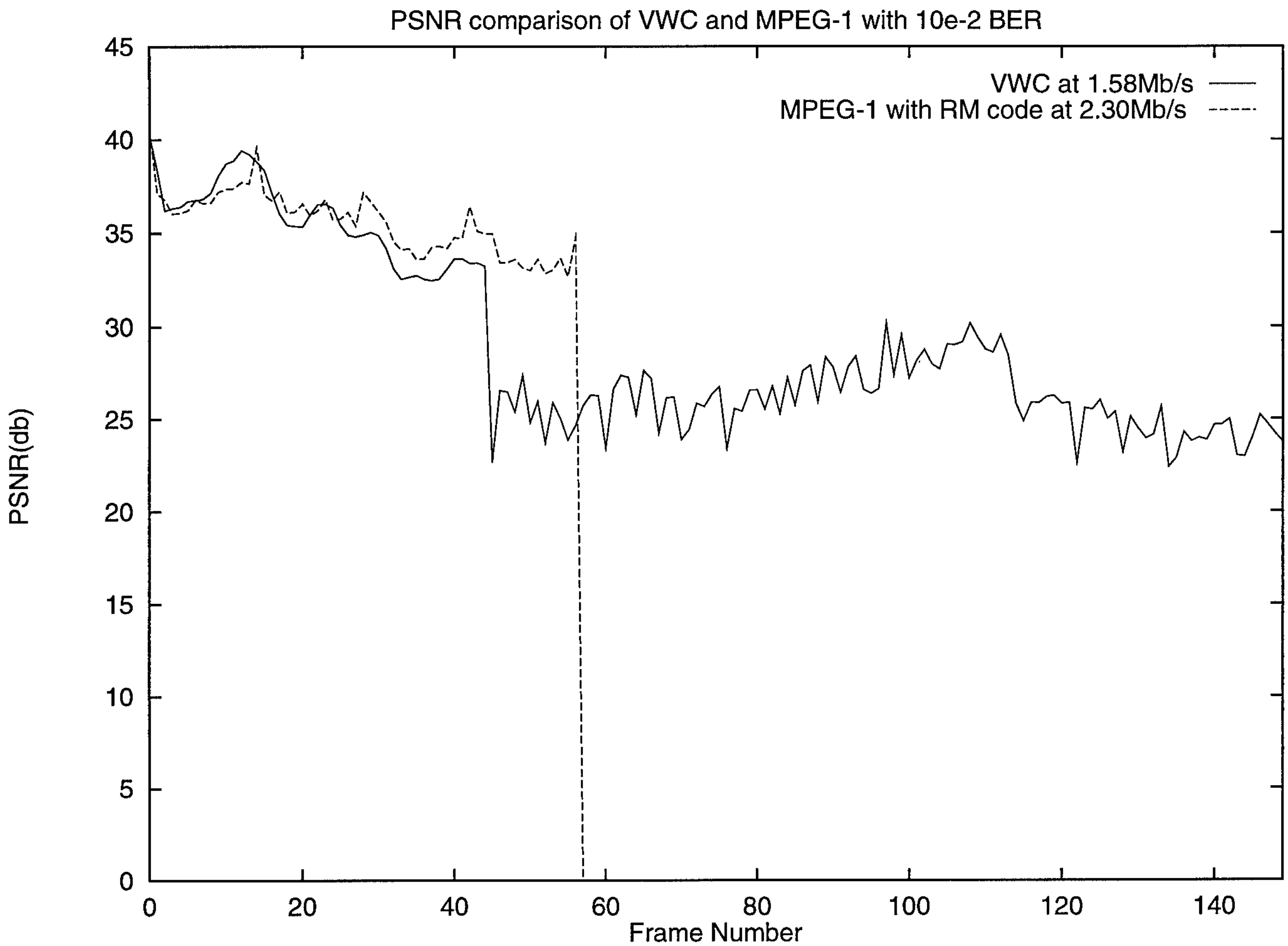

Figure 11: Comparison of VWC with MPEG-1 plus RM code at 2.3Mb/s and 10e-2 BER

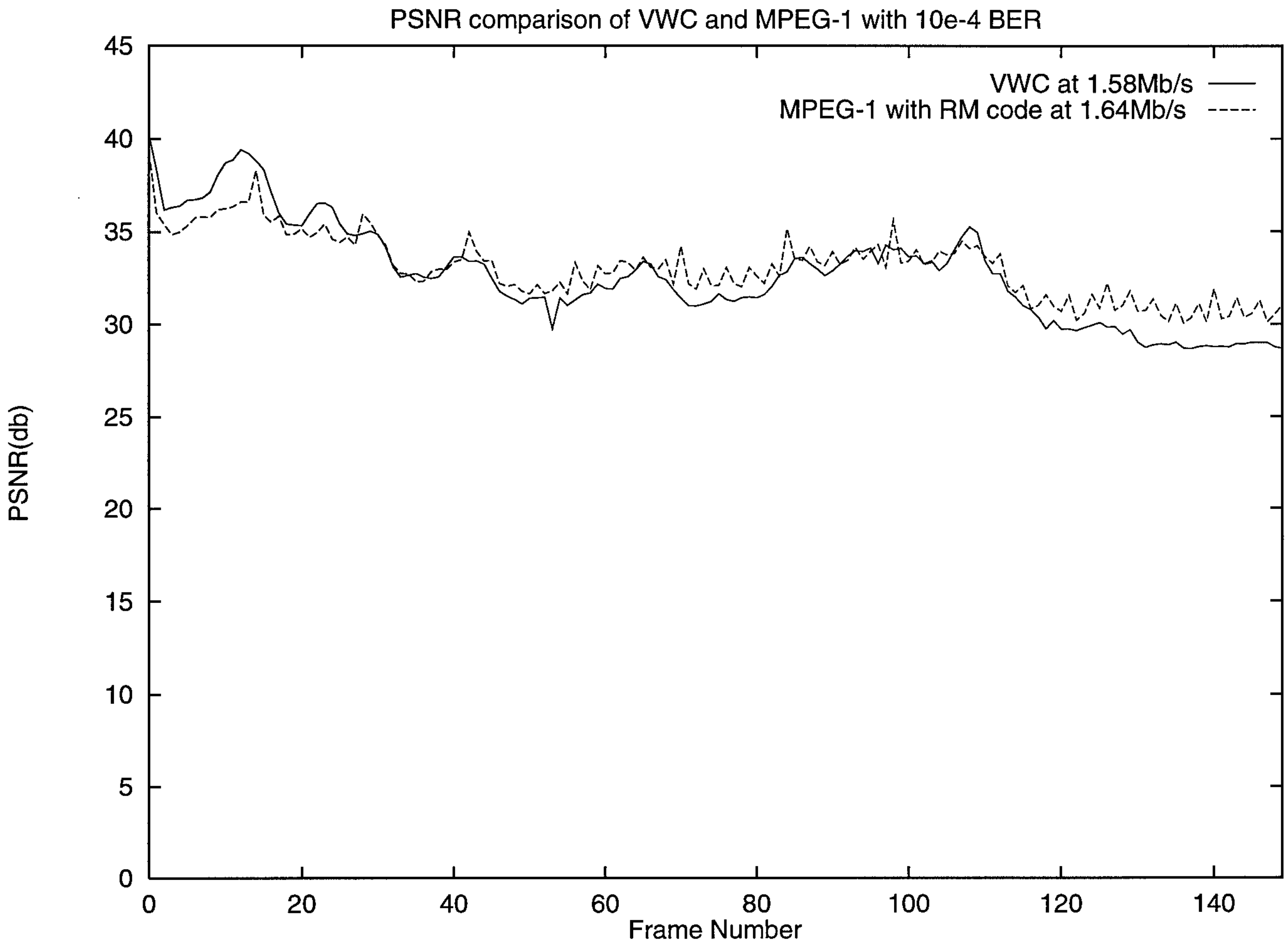

Figure 12: Comparison of VWC with MPEG-1 plus RM code at 1.64Mb/s and 10e-4 BER

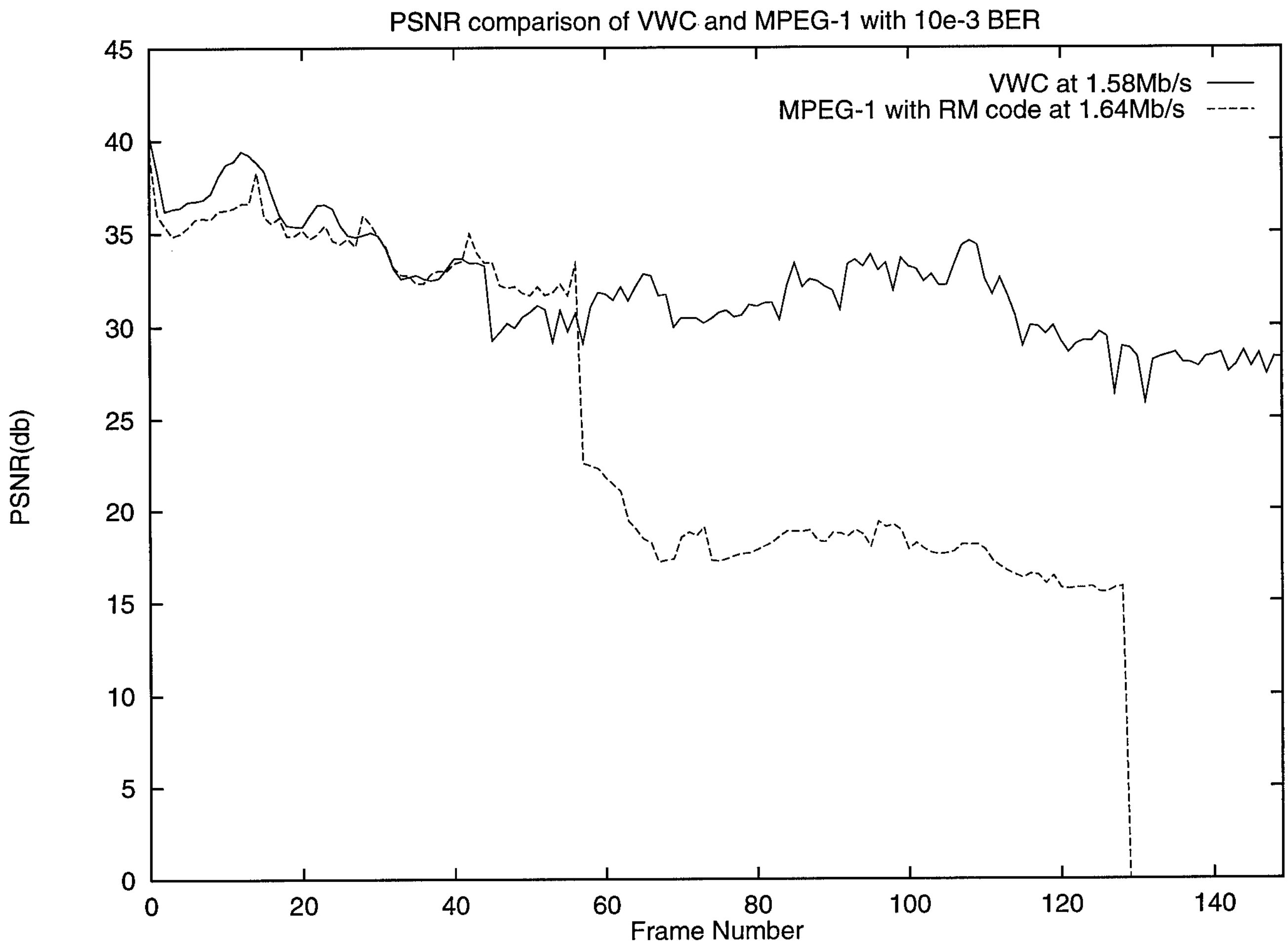

Figure 13: Comparison of VWC with MPEG-1 plus RM code at 1.64Mb/s and 10e-3 BER

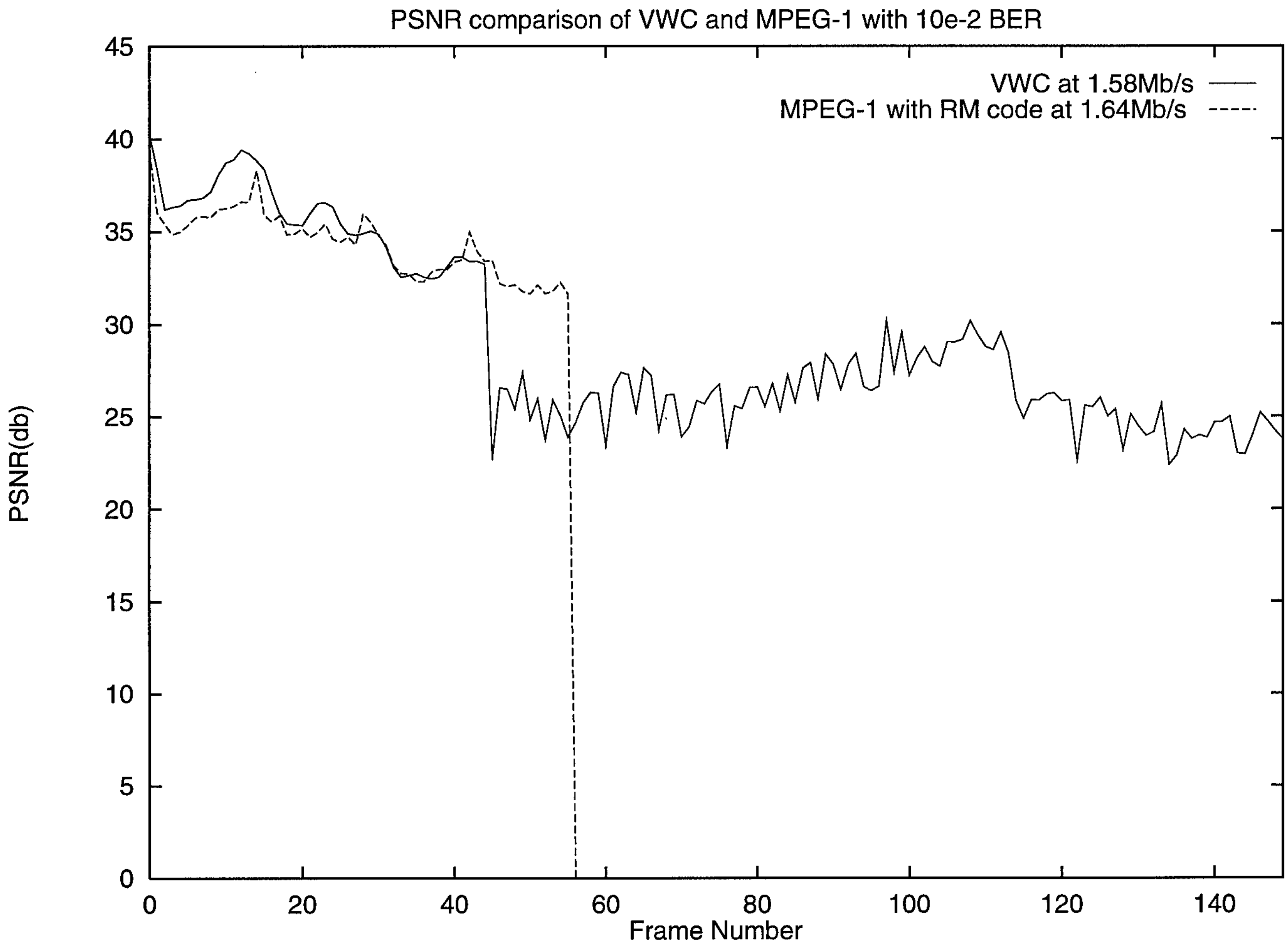

Figure 14: Comparison of VWC with MPEG-1 plus RM code at 1.64Mb/s and 10e-2 BER

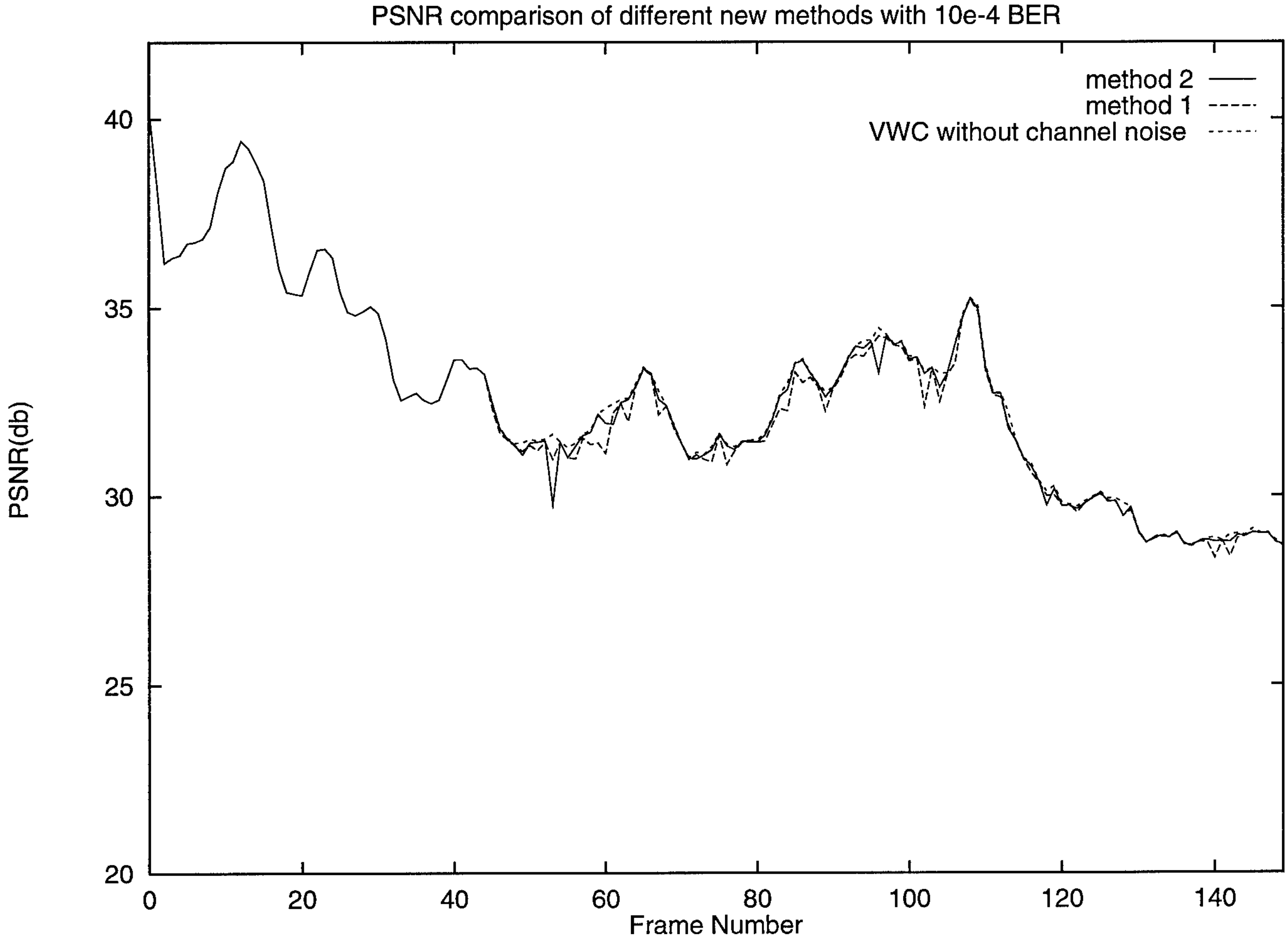

Figure 15: Comparison of relative performance of the new methods at 10e-4 BER

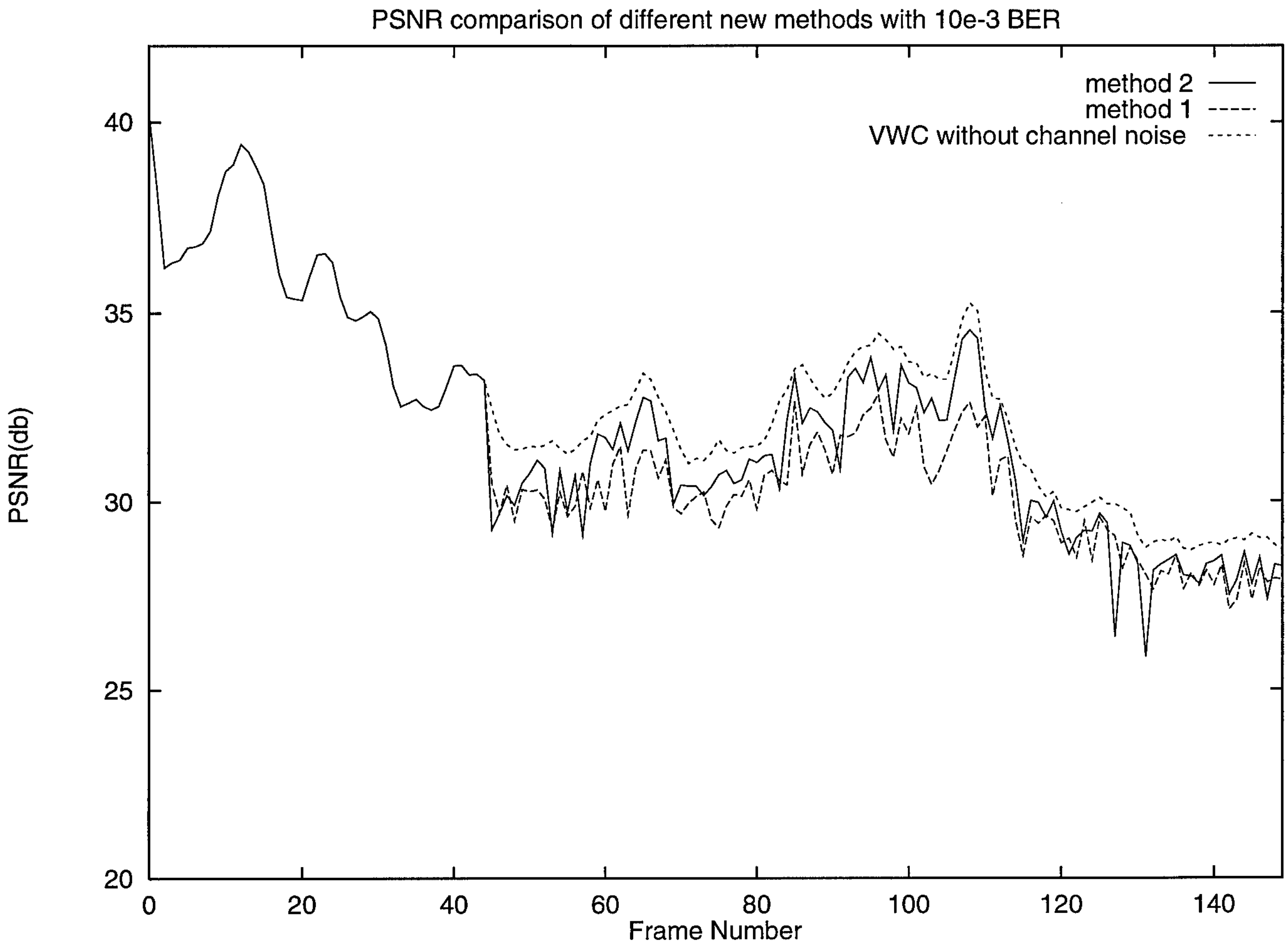

Figure 16: Comparison of relative performance of the new methods at 10e-3 BER

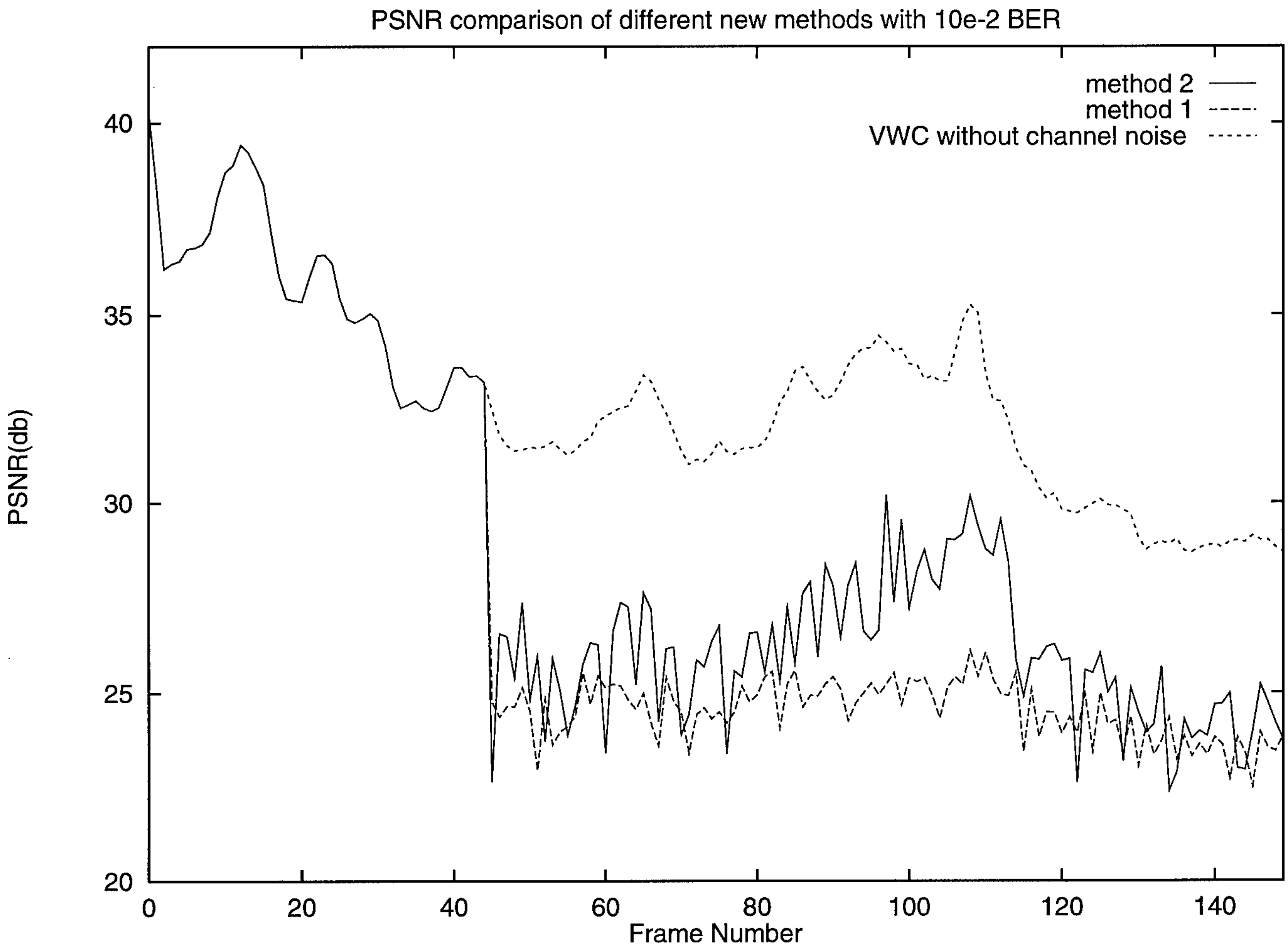

Figure 17: Comparison of relative performance of the new methods at 10e-2 BER